APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/808,899	03/15/2001	Maarten Koning	11283/31	4891

26646   7590   06/21/2004

KENYON & KENYON
ONE BROADWAY
NEW YORK, NY 10004

EXAMINER
ALI, SYED J

ART UNIT	PAPER NUMBER
2127	6

DATE MAILED: 06/21/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

		Application No.	Applicant(s)
Office Action Summary		09/808,899	KONING, MAARTEN
		Examiner	**Art Unit**
		Syed J Ali	2127

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) FROM
THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed
  after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
  Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any
  earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *15 March 2001*.
2a)☐ This action is **FINAL**.   2b)☒ This action is non-final.
3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is
   closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-28* is/are pending in the application.
   4a) Of the above claim(s) _____ is/are withdrawn from consideration.
5)☐ Claim(s) _____ is/are allowed.
6)☒ Claim(s) *1-28* is/are rejected.
7)☐ Claim(s) _____ is/are objected to.
8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.
10)☒ The drawing(s) filed on *08 June 2001* is/are: a)☒ accepted or b)☐ objected to by the Examiner.
   Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
   Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
   a)☐ All b)☐ Some * c)☐ None of:
   1.☐ Certified copies of the priority documents have been received.
   2.☐ Certified copies of the priority documents have been received in Application No. _____.
   3.☐ Copies of the certified copies of the priority documents have been received in this National Stage
      application from the International Bureau (PCT Rule 17.2(a)).
   * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
   Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413)
   Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1.   Claims 1-28 are pending in this application.

Claim Rejections - 35 USC § 102

2.   The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the

basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

3.   **Claims 27-28 are rejected under 35 U.S.C. 102(e) as being anticipated by Ninose et**

al. (USPN 6,105,098) (hereinafter Ninose).

4.   As per claim 27, Ninose teaches the invention as claimed, including a semaphore control

block associated with a semaphore, the semaphore control block comprising:

a holding task identification variable, the holding task identification variable configured

to indicate a task that presently holds the semaphore with which the semaphore control block is

associated (col. 8 lines 30-65);

a stealable variable, the stealable variable configured to indicate whether the semaphore

can be stolen from the task that presently holds the semaphore with which the semaphore control

block is associated (col. 3 lines 4-37).

5.   As per claim 28, Ninose teaches the invention as claimed, including the semaphore

control block associated with a semaphore according to claim 27, wherein the stealable variable

is a one-bit flag (col. 3 lines 4-37).

Claim Rejections - 35 USC § 103

6.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in
> section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are
> such that the subject matter as a whole would have been obvious at the time the invention was made to a person
> having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the
> manner in which the invention was made.

7.   **Claims 1-26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Morgan**

(USPN 6,721,948) in view of Ninose.

8.   As per claim 1, Morgan teaches the invention as claimed, including a method comprising:

       assigning a resource to a holding task (col. 4 lines 10-15; col. 4 line 59 - col. 5 line 15);

       receiving a request by a higher priority task to take the resource, the higher priority task

having higher priority than the holding task (col. 5 lines 16-22); and

       assigning the resource to the higher priority task (col. 5 lines 23-52; col. 6 lines 13-31).

9.   Ninose teaches the invention as claimed, including the following limitations not shown

by Morgan:

determining whether the holding task has used the resource since the resource was assigned to the holding task (col. 2 lines 7-11; col. 3 lines 4-13; col. 8 lines 30-39; col. 10 lines 16-23); and

releasing the resource when the higher priority task requests to take the resource and the holding task has not used the resource since the resource was assigned to the holding task (col. 2 lines 12-20; col. 3 lines 4-13; col. 14 line 42 - col. 15 line 49).

10.   It would have been obvious to one of ordinary skill in the art to combine Morgan and Ninose since the method of Morgan, while providing a framework for preempting a task, even while it is using a resource, thereby eliminating the need to save a resource's context when it is preempted. When the preempted task is resumed, a saved context is loaded that corresponds to the most recent watchpoint (or checkpoint). While this allows a preempted task to resume execution, changes to the resource in the time that the higher priority task was executing may result in inconsistent data, and incorrect processing results. Ninose remedies such a situation by setting a variable indicative of a resource's usage once a task has gained control of a resource. This prevents another task for preempting the resource while changes are potentially being made (col. 2 lines 7-20). Further, while Ninose does not explicitly state that preemption is enabled in favor of higher priority tasks, such is provided for, such as in cases of real-time systems that have hard deadlines. Ninose provides a means of monitoring a resource and aborting the ownership of a lock in case of a task holding a resource for an excessive period of time (col. 4 lines 55-62). Thus, Morgan and Ninose provide mutual benefits such that a task may be preempted in favor of higher priority task, while also protecting shared resources against inconsistent data.

11.   As per claim 2, Morgan teaches the invention as claimed, including a method comprising:

assigning a semaphore to a holding task (col. 4 lines 10-15; col. 4 line 59 - col. 5 line 15);

receiving a request by a higher priority task to take the semaphore, the higher priority

task having higher priority than the holding task (col. 5 lines 16-22); and

assigning the semaphore to the higher priority task (col. 5 lines 23-52; col. 6 lines 13-31).

12.   Ninose teaches the invention as claimed, including the following limitations not shown

by Morgan:

determining whether the holding task has executed since the semaphore was assigned to

the holding task (col. 2 lines 7-11; col. 3 lines 4-13; col. 8 lines 30-39; col. 10 lines 16-23); and

releasing the semaphore when the higher priority task requests to take the semaphore and

the holding task has not executed since the semaphore was assigned to the holding task (col. 2

lines 12-20; col. 3 lines 4-13; col. 14 line 42 - col. 15 line 49).

13.   It is noted that the shared resources in Morgan are not necessarily semaphores or mutual

exclusion semaphores.   The term *resource* as used by Morgan is may be any software or

hardware component, including shared tasks that perform a function, as well as memory

components or any device attached to the computer system (col. 1 lines 40-56; col. 2 lines 15-

52). Thus, hereinafter Morgan's method of locking shared tasks will be considered applicable to

any type of shared resource including mutexes or semaphores, as a provision is clearly made for

such a modification.

14.   As per claim 3, Morgan teaches the invention as claimed, including a method comprising:

assigning a semaphore to a holding task, the semaphore being a mutual exclusion semaphore (col. 4 lines 10-15; col. 4 line 59 - col. 5 line 15);

receiving a request by a higher priority task to take the semaphore, the higher priority task having higher priority than the holding task (col. 5 lines 16-22); and

assigning the semaphore to the higher priority task (col. 5 lines 23-52; col. 6 lines 13-31).

15.   Ninose teaches the invention as claimed, including the following limitations not shown by Morgan:

determining whether the holding task has executed since the semaphore was assigned to the holding task (col. 2 lines 7-11; col. 3 lines 4-13; col. 8 lines 30-39; col. 10 lines 16-23); and

releasing the semaphore held by the holding task when the higher priority task requests to take the semaphore and the holding task has not executed since the semaphore was assigned to the holding task (col. 2 lines 12-20; col. 3 lines 4-13; col. 14 line 42 - col. 15 line 49).

16.   As per claim 4, Ninose teaches the invention as claimed, including the method according to claim 3, wherein the step of determining whether the holding task has executed since the semaphore was assigned to the holding task includes testing a variable, the variable indicative of whether the holding task has executed since the semaphore was assigned to the holding task (col. 2 lines 7-11; col. 3 lines 4-13).

17.   As per claim 5, Ninose teaches the invention as claimed, including the method according to claim 4, wherein the variable is associated with the holding task (col. 8 lines 10-46).

18.   As per claim 6, Ninose teaches the invention as claimed, including the method according

to claim 4, wherein the variable is associated with the semaphore (col. 8 lines 10-46).

19.   As per claim 7, Ninose teaches the invention as claimed, including the method according

to claim 4, further comprising:

        setting the variable, when the semaphore is assigned to the holding task, to indicate that

the holding task has not executed since the semaphore was assigned to the holding task (col. 2

lines 7-11; col. 3 lines 4-13; col. 8 lines 30-39; col. 10 lines 16-23).

20.   As per claim 8, Morgan teaches the invention as claimed, including the method according

to claim 3, further comprising:

        assigning a second semaphore to a second holding task, the second semaphore being a

mutual exclusion semaphore (col. 1 lines 40-56; col. 4 lines 10-15; col. 4 line 59 - col. 5 line 15);

        receiving a request by a second higher priority task to take the semaphore, the second

higher priority task having higher priority than the second holding task (col. 5 lines 16-22);

21.   Ninose teaches the invention as claimed, including the following limitations not shown

by Morgan:

        determining whether the second holding task has executed since the second semaphore

was assigned to the second holding task (col. 2 lines 7-11; col. 3 lines 4-13; col. 8 lines 30-39;

col. 10 lines 16-23); and

maintaining control of the second semaphore by the second holding task when the second

higher priority task attempts to take the second semaphore and the second holding task has

executed since the second semaphore was assigned to the second holding task (col. 2 lines 12-20;

col. 3 lines 4-13; col. 8 lines 30-39; col. 14 line 42 - col. 15 line 49).

22.   As per claim 9, Ninose teaches the invention as claimed, including the method according

to claim 8, wherein the step of determining whether the second holding task has executed since

the second semaphore was assigned to the second holding task includes testing a second variable,

the second variable indicative of whether the second holding task has executed since receiving

the second semaphore (col. 2 lines 7-11; col. 3 lines 4-13).

23.   As per claim 10, Ninose teaches the invention as claimed, including the method

according to claim 9, further comprising:

setting the second variable to indicate that the second holding task has not executed when

the second semaphore is assigned to the second holding task (col. 2 lines 7-11; col. 3 lines 4-13;

col. 8 lines 30-39; col. 10 lines 16-23).

24.   As per claim 11, Ninose teaches the invention as claimed, including the method

according to claim 9, further comprising:

setting the second variable to indicate that the second holding task has executed when the

second holding task executes after receiving the second semaphore (col. 2 lines 7-11; col. 3 lines

4-13; col. 8 lines 30-39; col. 10 lines 16-23).

25.   As per claim 12, Ninose teaches the invention as claimed, including the method according to claim 9, wherein the second variable is associated with the second holding task (col. 8 lines 10-46).

26.   As per claim 13, Ninose teaches the invention as claimed, including the method according to claim 9, wherein the second variable is associated with the second semaphore (col. 8 lines 10-46).

27.   As per claim 14, Ninose teaches the invention as claimed, including the method of claim 3, further comprising:

    timing out a last request for the semaphore by the holding task if the last request would have already timed out had the holding task not received the semaphore by the time the semaphore is released (col. 4 lines 55-62).

28.   As per claim 15, Ninose teaches the invention as claimed, including the method of claim 3, further comprising:

    adding an entry for the holding task to a wait queue (col. 8 lines 30-65; col. 16 lines 4-18).

29.   As per claim 16, Morgan teaches the invention as claimed, including a method comprising:

assigning a semaphore to a holding task, the semaphore being a mutual exclusion

semaphore (col. 4 lines 10-15; col. 4 line 59 - col. 5 line 15);

receiving a request for the semaphore from a higher priority task, the higher priority task

having higher priority than the holding task (col. 5 lines 16-22);

assigning the semaphore to the higher priority task (col. 5 lines 23-52; col. 6 lines 13-31);

assigning a second semaphore to a second holding task, the second semaphore being a

mutual exclusion semaphore (col. 1 lines 40-56; col. 4 lines 10-15; col. 4 line 59 - col. 5 line 15);

and

receiving a request for the second semaphore from a second higher priority task, the

second higher priority task having higher priority than the second holding task (col. 5 lines 16-

22).

30.   Ninose teaches the invention as claimed, including the following limitations not shown

by Morgan:

setting a variable to indicate that the holding task has not executed since receiving the

semaphore when the holding task receives the semaphore, the variable indicative of whether the

holding task has executed since receiving the semaphore (col. 2 lines 7-11; col. 3 lines 4-13; col.

8 lines 30-39; col. 10 lines 16-23);

determining whether the holding task has executed since receiving the semaphore by

testing the variable (col. 2 lines 7-11; col. 3 lines 4-13; col. 8 lines 30-39; col. 10 lines 16-23);

releasing the semaphore held by the holding task when the higher priority task attempts to

take the semaphore and the holding task has not executed since receiving the semaphore (col. 2

lines 12-20; col. 3 lines 4-13; col. 14 line 42 - col. 15 line 49);

timing out a last request for the semaphore by the holding task if the last request would have timed out had the holding task not received the semaphore by the time the holding task releases the semaphore (col. 4 lines 55-62);

setting a second variable to indicate that the second holding task has not executed since receiving the second semaphore when the second holding task receives the second semaphore, the second variable indicative of whether the second holding task has executed since receiving the second semaphore (col. 2 lines 7-11; col. 3 lines 4-13; col. 8 lines 30-39; col. 10 lines 16-23);

setting the second variable to indicate the second holding task has executed since receiving the second semaphore, when the second holding task first executes after receiving the second semaphore (col. 2 lines 7-11; col. 3 lines 4-13; col. 8 lines 30-39; col. 10 lines 16-23);

determining whether the second holding task has executed since receiving the second semaphore by testing the second variable (col. 2 lines 7-11; col. 3 lines 4-13; col. 8 lines 30-39; col. 10 lines 16-23); and

maintaining control of the second semaphore by the second holding task when a second higher priority task attempts to take the semaphore and the second holding task has executed since receiving the second semaphore (col. 2 lines 12-20; col. 3 lines 4-13; col. 8 lines 30-39; col. 14 line 42 - col. 15 line 49).

31.   As per claim 17, Morgan teaches the invention as claimed, including an article of manufacture comprising a computer-readable medium having stored thereon instructions adapted to be executed by a processor, the instructions which, when executed, define a series of steps to be used to control a method for resource control, said steps comprising:

assigning a semaphore to a holding task, the semaphore being a mutual exclusion semaphore (col. 4 lines 10-15; col. 4 line 59 - col. 5 line 15);

receiving a request by a higher priority task to take the semaphore, the higher priority task having higher priority than the holding task (col. 5 lines 16-22); and

assigning the semaphore to the higher priority task (col. 5 lines 23-52; col. 6 lines 13-31).

32.   Ninose teaches the invention as claimed, including the following limitations not shown by Morgan:

determining whether the holding task has executed since the semaphore was assigned to the holding task (col. 2 lines 7-11; col. 3 lines 4-13; col. 8 lines 30-39; col. 10 lines 16-23); and

releasing the semaphore held by the holding task when the higher priority task requests to take the semaphore and the holding task has not executed since the semaphore was assigned to the holding task (col. 2 lines 12-20; col. 3 lines 4-13; col. 14 line 42 - col. 15 line 49).

33.   As per claim 18, Morgan teaches the invention as claimed, including an article of manufacture comprising a computer-readable medium having stored thereon instructions adapted to be executed by a processor, the instructions which, when executed, define a series of steps to be used to control a method for resource control, said steps comprising:

assigning a semaphore to a holding task, the semaphore being a mutual exclusion semaphore (col. 4 lines 10-15; col. 4 line 59 - col. 5 line 15);

receiving a request for the semaphore from a higher priority task, the higher priority task having higher priority than the holding task (col. 5 lines 16-22);

assigning the semaphore to the higher priority task (col. 5 lines 23-52; col. 6 lines 13-31);

assigning a second semaphore to a second holding task, the second semaphore being a

mutual exclusion semaphore (col. 1 lines 40-56; col. 4 lines 10-15; col. 4 line 59 - col. 5 line 15);

and

receiving a request for the second semaphore from a second higher priority task, the

second higher priority task having higher priority than the second holding task (col. 5 lines 16-

22).

34.   Ninose teaches the invention as claimed, including the following limitations not shown

by Morgan:

setting a variable to indicate that the holding task has not executed since receiving the

semaphore when the holding task receives the semaphore, the variable indicative of whether the

holding task has executed since receiving the semaphore (col. 2 lines 7-11; col. 3 lines 4-13; col.

8 lines 30-39; col. 10 lines 16-23);

determining whether the holding task has executed since receiving the semaphore by

testing the variable (col. 2 lines 7-11; col. 3 lines 4-13; col. 8 lines 30-39; col. 10 lines 16-23);

releasing the semaphore held by the holding task when the higher priority task attempts to

take the semaphore and the holding task has not executed since receiving the semaphore (col. 2

lines 12-20; col. 3 lines 4-13; col. 14 line 42 - col. 15 line 49);

timing out a last request for the semaphore by the holding task if the last request would

have timed out had the holding task not received the semaphore by the time the holding task

releases the semaphore (col. 4 lines 55-62);

setting a second variable to indicate that the second holding task has not executed since

receiving the second semaphore when the second holding task receives the second semaphore,

the second variable indicative of whether the second holding task has executed since receiving

the second semaphore (col. 2 lines 7-11; col. 3 lines 4-13; col. 8 lines 30-39; col. 10 lines 16-23);

setting the second variable to indicate the second holding task has executed since

receiving the second semaphore, when the second holding task first executes after receiving the

second semaphore (col. 2 lines 7-11; col. 3 lines 4-13; col. 8 lines 30-39; col. 10 lines 16-23);

determining whether the second holding task has executed since receiving the second

semaphore by testing the second variable (col. 2 lines 7-11; col. 3 lines 4-13; col. 8 lines 30-39;

col. 10 lines 16-23); and

maintaining control of the second semaphore by the second holding task when a second

higher priority task attempts to take the semaphore and the second holding task has executed

since receiving the second semaphore (col. 2 lines 12-20; col. 3 lines 4-13; col. 8 lines 30-39;

col. 14 line 42 - col. 15 line 49).

35.   As per claim 19, Morgan teaches the invention as claimed, including a system,

comprising:

a semaphore (col. 4 lines 10-15; col. 4 line 59 - col. 5 line 15); and

a semaphore control mechanism configured to release the semaphore if

(a) a first task holds the semaphore (col. 4 line 59 - col. 5 line 22), and

(b) a second task having a higher priority than the first task attempts to take the

semaphore (col. 4 line 59 - col. 5 line 22).

36.   Ninose teaches the invention as claimed, including the following limitations not shown
by Morgan:

a semaphore control mechanism configured to release the semaphore if

(c) when the second task attempts to take the semaphore, the first task has not
executed since receiving the semaphore (col. 2 lines 12-20; col. 3 lines 4-13; col. 14 line
42 - col. 15 line 49).

37.   As per claim 20, Morgan teaches the invention as claimed, including a system,
comprising:

a semaphore, the semaphore being a mutual exclusion semaphore (col. 4 lines 10-15; col.
4 line 59 - col. 5 line 15); and

a semaphore control mechanism, the semaphore control mechanism configured to release
the semaphore if

(a) a first task holds the semaphore (col. 4 line 59 - col. 5 line 22), and

(b) a second task having higher priority than the first task attempts to take the
semaphore (col. 4 line 59 - col. 5 line 22).

38.   Ninose teaches the invention as claimed, including the following limitations not shown
by Morgan:

a semaphore control mechanism, the semaphore control mechanism configured to release
the semaphore if

(c) when the second task attempts to take the semaphore, the first task has not

executed since receiving the semaphore (col. 2 lines 12-20; col. 3 lines 4-13; col. 14 line

42 - col. 15 line 49).

39.   As per claim 21, Ninose teaches the invention as claimed, including the system according

to claim 20, wherein the semaphore control mechanism is configured not to release the

semaphore when the second task attempts to take the semaphore and the first task has executed

since receiving the semaphore (col. 2 lines 12-20; col. 3 lines 4-13; col. 8 lines 30-39; col. 14

line 42 - col. 15 line 49).

40.   As per claim 22, Ninose teaches the invention as claimed, including the system according

to claim 20, further comprising:

    a variable indicative of whether the first task has executed since receiving the semaphore

(col. 2 lines 7-11; col. 3 lines 4-13).

41.   As per claim 23, Ninose teaches the invention as claimed, including the system according

to claim 22, wherein the variable is associated with the semaphore (col. 8 lines 10-46).

42.   As per claim 24, Ninose teaches the invention as claimed, including the system according

to claim 22, wherein the variable is associated with the first task (col. 8 lines 10-46).

43.   As per claim 25, Ninose teaches the invention as claimed, including the system according

to 20, further comprising:

        a timeout mechanism, the timeout mechanism configured to time out a last request by the

first task for the semaphore if the second task attempts to take the semaphore and the first task

has not executed since receiving the semaphore and the last request would have timed out had

the first task not received the semaphore by the time the semaphore is released (col. 4 lines 55-

62).

44.   As per claim 26, Morgan teaches the invention as claimed, including a system,

comprising:

        a semaphore, the semaphore being a mutual exclusion semaphore (col. 4 lines 10-15; col.

4 line 59 - col. 5 line 15);

        a first task, the first task holding the semaphore (col. 4 lines 10-15; col. 4 line 59 - col. 5

line 15); and

        a second task, the second task having higher priority than the first task (col. 1 lines 40-56;

col. 4 lines 10-15; col. 4 line 59 - col. 5 line 15).

45.   Ninose teaches the invention as claimed, including the following limitations not shown

by Morgan:

        a variable indicative of whether the first task has executed since receiving the semaphore,

the variable associated with the first task (col. 2 lines 7-11; col. 3 lines 4-13; col. 8 lines 30-39;

col. 10 lines 16-23);

a semaphore control mechanism configured to release the semaphore when the second task attempts to take the semaphore and the first task has not executed since receiving the semaphore and not to release the semaphore when the second task attempts to take the semaphore and the first task has executed since receiving the semaphore (col. 2 lines 12-20; col. 3 lines 4-13; col. 14 line 42 - col. 15 line 49); and

a timeout mechanism, the timeout mechanism configured to time out a last request by the first task for the semaphore if the second task attempts to take the semaphore and the first task has not executed since receiving the semaphore and the last request by the first task for the semaphore would have timed out had the first task not received the semaphore by the time the semaphore is released (col. 4 lines 55-62).

Conclusion

46.   The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Insalaco et al. (USPN 5,305,448) teaches a "lock stealing" mechanism that allows a shared resource to be reclaimed when some sort of system error has occurred.

Van Dort (USPN 6,148,003) teaches indicators that relate to the usage of a shared resource including the priority of the owner of the resource.

Presler-Marshall (USPN 6,199,094) teaches an indicator that is set when a resource is used.

Nakahara et al. (USPN 6,253,225) teaches a method of enabling preemption of a shared resource, such as in accordance with a task's priority.

Sorace et al. (USPN 6,477,597) teaches indicators for showing that a task has obtained a lock on a resource without using it.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Syed J Ali whose telephone number is (703) 305-8106. The examiner can normally be reached on Mon-Fri 8-5:30, 2nd Friday off.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng-Ai T An can be reached on (703) 305-9678. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Syed Ali
June 2, 2004

MENG-AL T. AN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100